SCSI-3 Fault Tolerant Controller Configurations

utilizing SCC & New Event Codes

Edited by

Steve Sicola

31-August-1995

Purpose

The purpose of this document is to profile the use of SCC with new ASC/ASCQ's that would allow Operating System Drivers to be written to generically support fault tolerant controller configurations in open systems environments. This document further describes the implicit assumptions about the relationship between controllers in a fault tolerant controller configurations so that deviations from the SCC model are not needed.

Overview

Within the confines of the SCSI-3 SCC (Standard Controller Commands) lie the tools in order to 'register' and 'report' the existence of fault tolerant controller configurations that contain two or more controllers sharing the same storage devices and paths to those storage devices. The SACL within each controller in a fault tolerant configuration is logically presented as a single SACL with multiple ports. Specifically, the two controllers must present the exact same configuration when SCC 'Report' commands are presented to either controller.

Assumptions

The single controller with multi-ports works based upon two assumptions that are typically no problem for controllers in fault tolerant configurations to comply with. These assumptions are

1. Controllers in a fault tolerant configuration communicate with each other to relate changes in state or configuration
2. Controllers in a fault tolerant configuration will always have report the same configuration to all attached hosts.

3. Controllers supported will include those with single or multiple host interfaces, single or multiple shared device interfaces, and those that contain separate or integrated communication between controllers in a fault tolerant configuration.

In order for controllers to be in a fault tolerant configuration, this assumption is a very logical conclusion so that failover can be graceful, so as not to cause data corruption upon failover, reduce the time of failover (time to LUNs appearing on the alternate controller).

Functional Description

Utilizing SCC, the Attach Component Device command will be used by host to create an attachment between controllers to create or add to a fault tolerant controller configuration. The configuration may also be setup by other means available to the controllers (external user interfaces) in which case the use of Create Attachment command is unnecessary.

In order for host computers to recognize fault tolerant controller configurations, the Report Component Device Attachment command must be used to interrogate controllers to determine that multiple paths do exist to the same storage devices.

The controllers must also share the use of LUN0 on every controller in the same controller configuration. LUN0 on each controller will report the same configuration. This keeps the subsystem consistent. Furthermore, any host or external user interface configurations entered on one controller MUST also be relayed immediately to all other controllers in the same fault tolerant controller configuration.

The Attach Component command needs one more aspect (noted below in open issues) in order to adequately cover all possible actions of a controllers in a fault tolerant configuration. That aspect is controller configuration naming. The naming must cover the needs of the overall system the configuration is attached to. The naming must handle a change in membership, either from an addition to the configuration, a deletion from the configuration, or from a replacement in configuration (after failure). The name must be unique with a system installation, but not necessarily world-wide unique. The name must essentially be a controller configuration ‘handle’ that can be used by any host operation system to key off of in order to handle multiple paths to the same devices or LUNs.

The events of failover and failback are defined as follows:

1. Failover - The failure of a controller to continue accessing of attached devices for whatever reason resulting in the failure to serve host requests. Failover to a surviving controller means that the surviving controller must be able to restore access to attached devices and restore request service to hosts. The action in some cases is required to be as quick as possible.

2. Failback - The restoration to active service by a previously failure of a controller. The controller that restored access originally for the failed controller may signal attached hosts that the resource has returned and is now available for service and will continue to service the attached devices unless directed otherwise (reserve and release activities).

The failover and failback events noted above can be achieved in many ways within the controller and are outside the scope of this profile, except in the area of event codes denoted the occurrence of the failover or failback event. In these cases, two new ASC/ASCQ's are required to support this possibility. These two event codes would denote Failover and Failback. They would be used by the controller taking over for a failed controller or that the failed controller has returned. The uses for these event codes are to allow for much more pro-active host involvement by port drivers instead of the use of timeouts. Timeouts based event actions do work adequately, but in some cases do NOT meet the needs of highly available customer installations. The needs in these cases are speedy failover and speedy failback in order to regain access in the event of failure and to regain performance in the event of controller replacement/re-initialization.

Additional Requirements & Issues for SCSI-3

The event codes would be used with Unit Attentions or AENs from the chosen 'Failover/Failback' Controller. The choice of which controller in a multi-controller fault tolerant configuration is outside the scope of this profile because the mechanisms to allow choice are here with the use of the new event codes.

The unique identification of a Controller Configuration is required. It does not need to be a World-wide type name, rather a 'system-wide' unique name that can cover the configuration across the hosts to which it is attached. This name will survive any controller failures and replacements, so as to not rely upon the serial number of the controller or any packaging specific addresses. Hosts will 'know' about the Controller configuration by name, simplifying any mapping between multiple access paths to the same data.